

Applying full polarization A-Projection to very wide field of view instruments: An imager for LOFAR

C. Tasse^{1,2,3}, B. van der Tol⁴, J. van Zwieten⁵, Ger van Diepen⁵, and S. Bhatnagar⁶

¹ GEPI, Observatoire de Paris, CNRS, Université Paris Diderot, 5 place Jules Janssen, 92190 Meudon, France

² Department of Physics & Electronics, Rhodes University, PO Box 94, Grahamstown, 6140, South Africa

³ SKA South Africa, 3rd Floor, The Park, Park Road, Pinelands, 7405, South Africa

⁴ Sterrewacht Leiden, PO Box 9513, 2300 RA, Leiden, The Netherlands

⁵ Netherlands Institute for Radio Astronomy (ASTRON), Postbus 2, 7990 AA Dwingeloo, The Netherlands

⁶ National Radio Astronomy Observatory, Socorro, NM 87801, USA

Received [date] / Accepted [date]

Abstract. The aimed high sensitivities and large fields of view of the new generation of interferometers impose to reach high dynamic range of order $\sim 1:10^6$ to $1:10^8$ in the case of the Square Kilometer Array. The main problem is the calibration and correction of the Direction Dependent Effects (DDE) that can affect the electromagnetic field (antenna beams, ionosphere, Faraday rotation, etc.). As shown earlier the A-Projection is a fast and accurate algorithm that can potentially correct for any given DDE in the imaging step. With its very wide field of view, low operating frequency ($\sim 30 - 250$ MHz), long baselines, and complex station-dependent beam patterns, the Low Frequency Array (LOFAR) is certainly the most complex SKA precursor. In this paper we present a few implementations of A-Projection applied to LOFAR that can deal with non-unitary station beams and non-diagonal Mueller matrices. The algorithm is designed to correct for all the DDE, including individual antenna, projection of the dipoles on the sky, beam forming and ionospheric effects. We describe a few important algorithmic optimizations related to LOFAR's architecture allowing us to build a fast imager. Based on simulated datasets we show that A-Projection can give dramatic dynamic range improvement for both phased array beams and ionospheric effects. We will use this algorithm for the construction of the deepest extragalactic surveys, comprising hundreds of days of integration.

1. Introduction: LOFAR and the direction dependent effects

With the building or development of many large radio telescopes (LOFAR, EVLA, ASKAP, MeerKAT, MWA, SKA, e-Merlin), radio astronomy is undergoing a period of rapid development. New issues arise with the development of these new types of interferometer, and the approximations applicable to the older generation of instruments are not valid anymore. Specifically, they have wide fields of view and will be seriously affected by the Direction Dependent Effects (DDE). Dealing with the DDE in the framework of calibration and imaging represents an unavoidable challenge, on both the theoretical, numerical and technical aspects of the problem (see Bhatnagar 2009, for a detailed review of the problems associated with calibration and wide field imaging in the presence of DDE).

This is particularly true for the Low Frequency Array (LOFAR). It is an instrument that observes in a mostly unexplored frequency range ($\nu \lesssim 240$ MHz), and will be one of the largest radio telescopes ever built in terms of collecting area. LOFAR's design is built on a combina-

tion of phased array and interferometer (see de Vos et al. (2009) for a description of the LOFAR system). It is made of 40 stations in the Netherlands, and 8 international stations (5 in Germany, 1 in France, England, and Sweden). The High Band Antenna stations (110-240 MHz, HBA hereafter) are made of 24 to 96 *tiles* of 4×4 antenna coherently summed, while the Low Band Antenna (10-80 MHz, LBA) are clustered in groups of 96 elements. At the station level, the signals from the individual antennas or tiles (in the cases of LBA and HBA respectively) are phased and summed by the *beamformer*. This step amounts to forming a virtual antenna pointing at the targeted field location. The data is transported from the various stations of the LOFAR array to the correlator. The whole process and the pipeline architecture have been described in more details in Heald et al. (2010). LOFAR is affected by many complex baseline-time-frequency⁴ dependent DDE, consisting mainly of the antenna/station beams and the ionosphere, which varies on angular scales of degrees and time scales of ~ 10 minutes and ~ 30 seconds respectively. We cur-

rently have models of both the high-band and low-band station beams (HBA and LBA respectively).

As shown in Bhatnagar et al. (2008) A-Projection allows to estimate sky images, taking into account all the possible complicated effects associated to the DDE (see also Bernardi et al. 2011; Mitchell et al. 2012; Sullivan et al. 2012, in the context of the Murchison Widefield Array and forward modeling). However contrarily to dishes-based interferometers, where the beam shape and polarization angle are affected by pointing errors and rotated on the sky by the parallactic angle (depending on the dish mount), LOFAR is based on phased arrays that have very wide fields of view (up to ~ 12 degrees), non-trivial and quickly varying beams, thereby driving complicated polarization effects. Technically speaking, the very wide fields of view instruments that aim to reach high dynamic range, have to deal with baseline-dependent non-diagonal Mueller matrices (see Sec. 2 for a detailed discussion). For the VLA implementation, due to the approximate *Unitarity* of VLA beams, it was sufficient for A-Projection to take into account the diagonal terms of the Mueller matrices to demonstrate corrections for instrumental polarization. That is not possible for LOFAR that has heavily non-diagonal baseline-associated Mueller matrices, and all 4×4 Mueller terms have to be taken into account.

We show in this paper that the scheme described in Bhatnagar et al. (2008) can indeed deal with the heavily non-unitary beams associated with the very wide field of view of phased arrays. Our imaging algorithm could take as input any model or calibration solution or ionosphere phase screen. In Sec. 2 we describe the issues related with the usage of phased arrays in interferometers, and focus on the LOFAR-related issues *i.e.* the polarization aspects and baseline dependence of the DDE. We describe a few important algorithmic optimizations related to LOFAR's architecture allowing us to build a fast imager¹. In Sec. 3 we describe the A-Projection algorithm first presented in Bhatnagar et al. (2008), and detail the various implementations and optimizations we have found to make it reasonably fast in the case of LOFAR. We present the results in Sec. 4 and show that beam and ionosphere corrections can both be performed at high accuracy. We summarize and discuss the next developments in Sec. 5.

2. Polarization effects associated with very wide fields of view interferometers

In this section we describe the polarizations effects associated with the complex structure of the DDE inherent to the usage of phased arrays that have very wide fields of view and non-diagonal baseline-dependent Mueller matrices (or non-unitary Jones matrices in the case of interferometers having similar antennas, see Fig. 1 and the discussion in Sec. 2.1).

With its long baselines, large fractional bandwidth, very wide field of view and station-dependent effective Jones matrices (beam, ionosphere, Faraday rotation), the Mueller matrices to be considered are not only non-diagonal, but are also baseline-dependent. In order to highlight some of the main complications associated with very wide fields of view instruments, in Sec. 2.1 we describe in detail the structure of the linear operator introduced by Bhatnagar et al. (2008). We propose in Sec. 2.2 a method to approximately correct for the associated effects corrupting the image plane (see also Rau & Cornwell 2011, for other examples of the use of linear operator in the context of image synthesis and deconvolution).

2.1. Description of the baseline-dependence, DDE and polarization effects

For convenience, in this section and throughout the paper, we do not show the sky term $\sqrt{1-l^2-m^2}$ that usually divide the sky to account for the projection of the celestial sphere on the plane, as this has no influence on the results. The DDE below are baseline-dependent, as it is the case for LOFAR, and the Measurement Equation formalism can properly model those effects (for extensive discussions on validity and limitations of the Measurement Equation see Hamaker et al. 1996; Smirnov 2011, and see Appendix A for a short review of our needs). If V_{pq}^{meas} is the set of 4-polarization measurement (XX, XY, YX, YY), G_p is the direction independent Jones matrix of antenna- p , then the corrected visibilities on baseline pq are $V_{pq}^{corr} = [G_{tv,p}]^{-1} \cdot V_{pq}^{meas} \cdot [G_{tv,q}^H]^{-1}$, and we can write:

$$\text{Vec}(V_{pq}^{corr}) = \int_S (D_{q,s}^{tv,*} \otimes D_{p,s}^{tv}) \cdot \text{Vec}(I_s) \cdot \exp(-2i\pi\phi(u, v, w, s)) ds \quad (1)$$

where I is the 4-polarization sky, \otimes is the Kronecker product, Vec is the operator¹¹ that transforms a 2×2 matrix into a dimension 4 vector, and $\phi(u, v, w, s) = u.l + v.m + w.(\sqrt{1-l^2-m^2}-1)$ models the product of the effects of correlator, sky brightness and array geometry. The matrix $D_{q,s}^{tv,*} \otimes D_{p,s}^{tv}$ is a 4×4 matrix, and throughout the text we refer to it as the *Mueller* matrix². We can also write Eq. 1 in terms a series of linear transformations:

$$V_{pq}^{t,\nu} = W_{pq}^{t,\nu} \cdot S_{pq}^{t,\nu} \cdot F \cdot \mathcal{D}_{pq}^{t,\nu} \cdot I \quad (2)$$

where $V_{pq}^{t,\nu}$ are the $4N_{pix}^{t,\nu}$ visibility measurement points in the time-frequency block in which the direction dependent effects are assumed to be constant. If N_{pix} is the number of pixels in the sky model, the *true* sky image vector I has a size of $4N_{pix}$ and contains the full polarization information (XX _{x} , XY _{x} , YX _{x} , YY _{x}) on the x^{th} pixel at the $4x$ position. $\mathcal{D}_{pq}^{t,\nu}$ contains the direction dependent effects, and is a $(4N_{pix}^{t,\nu}) \times (4N_{pix})$ block diagonal matrix.

¹ Our software (*awimager*) is built on the Casa imager implementation.

² This is not completely true as traditionally the Mueller matrix multiplies an (I, Q, U, V) vector and not an (XX, XY, YX, YY) correlation vector.

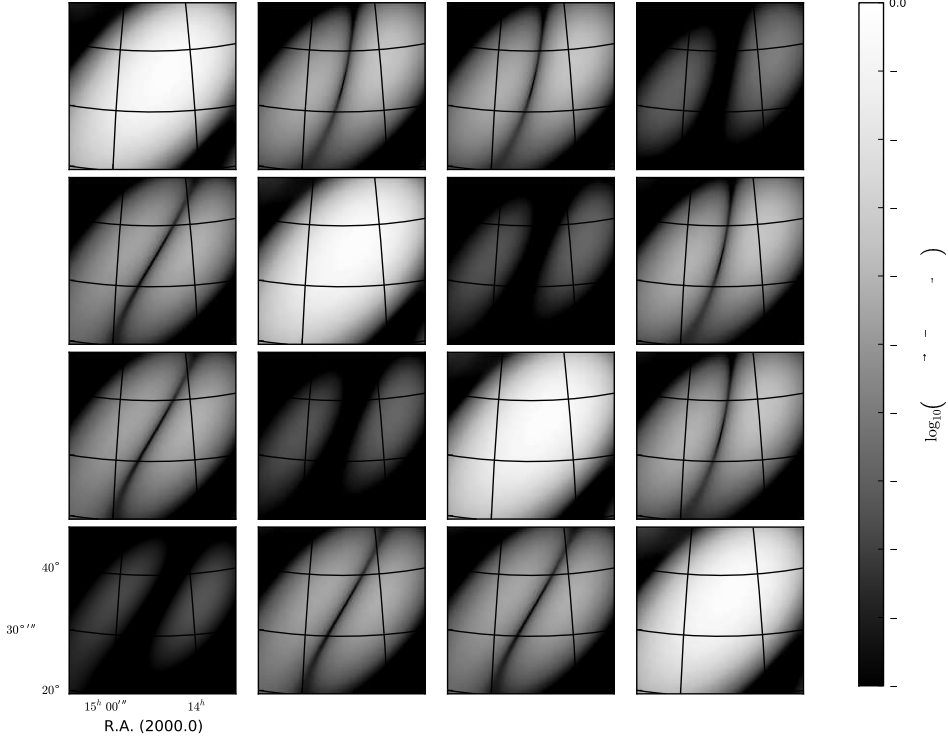


Fig. 1. LOFAR stations are phased arrays characterized by a large field of view. The X/Y polarimetric measurements are therefore non trivial standard compared to a radio telescope using dishes: we have to take into account the projection of the dipoles that are generally non-orthogonal on the sky and the fact that this angle varies across the field of view. This figure shows the Mueller matrix corresponding to baseline (01) in a given time and frequency slot, normalized by the Mueller matrix at the phase center (see text). Each pixel in the plot (i, j) shows the amplitude of the (i, j) Mueller matrix element in a certain direction s using a logarithmic scale. Even in this normalized version, the off-diagonal Mueller terms are as high as 10% and cannot be neglected.

On a given baseline (p, q) , each of its 4×4 block is the $\mathcal{D}_{pq}^{t, \nu}(\mathbf{s}_x) = D_q^*(\mathbf{s}_x) \otimes D_p(\mathbf{s}_x)$ Mueller matrix evaluated at the location of the x^{th} pixel. F is the Fourier transform operator of $(4N_{pix}) \times (4N_{pix})$. Each of its (4×4) block is a scalar matrix, the scalar being the kernel of the Fourier basis $\exp(-2i\pi\phi(u, v, w, \mathbf{s}))$. The matrix $S_{pq}^{t, \nu}$ is the uv-plane sampling function for that visibility of size $4N_{pq}^{t, \nu} \times 4N_{pix}$, and $W_{pq}^{t, \nu}$ is the diagonal $4N_{pq}^{t, \nu} \times 4N_{pq}^{t, \nu}$ matrix containing the weights associated with the $4N_{pq}^{t, \nu}$ visibilities.

We show in Fig. 1 that the Mueller matrix is non-diagonal for LOFAR. The various panels show the amplitude of the direction dependent Mueller matrix using our beam model only (therefore it does not include Faraday rotation and ionosphere), for the baseline $(p, q) = (0, 1)$ in a given time and frequency slot. In order to minimize the off-diagonal elements, we have computed $\mathcal{D}_0(\mathbf{s}) = [\mathcal{D}(\mathbf{s}_0)]^{-1} \cdot \mathcal{D}(\mathbf{s})$ in the direction \mathbf{s} , where \mathbf{s}_0 is the phase center direction. Intuitively, the normalization of the Mueller matrix by $\mathcal{D}(\mathbf{s}_0)$ makes the projected dipoles on the sky orthogonal at the phase center (off-diagonal elements are zero there), while this gets less true as we go further from the targeted location. The off-diagonal ele-

ments can be as high as $\sim 10\%$ a few degrees from the pointed location and contrarily to most interferometers they cannot be neglected in the case of LOFAR.

We are generally interested in using the total set of visibilities over baselines, time and frequencies, having $4N_V$ points. We can write:

$$V = A \cdot I + \epsilon \quad (3)$$

where ϵ is noise, and A is a $(4N_V) \times (4N_{pix})$ matrix, made of the $W_{pq}^{t, \nu} \cdot S_{pq}^{t, \nu} \cdot F \cdot \mathcal{D}_{pq}^{t, \nu}$ on top of each other (each have dimension $4N_{block} \times 4N_{pix}$, where N_{block} is the total number of time-frequency blocks). We can also write $A = \mathcal{W} \mathcal{S} \mathcal{F} \mathcal{D}$, where \mathcal{D} has size $4N_{blocks} N_{pix} \times N_{pix}$, and has all the $\mathcal{D}_{pq}^{t, \nu}$ on top of each other. \mathcal{F} is the block-diagonal Fourier transform operator with size $4N_{blocks} N_{pix} \times 4N_{blocks} N_{pix}$, with all of its blocks $4N_{pix} \times 4N_{pix}$ being equal to the matrix F appearing in Eq. 2. \mathcal{S} and \mathcal{W} are the sampling and weights matrix of sizes $(4N_V) \times (4N_{blocks} N_{pix})$ and $(4N_V) \times (4N_V)$ respectively. The transformation of Eq. 3 is *exact*. Estimating a sky from a sparsely sampled measured set of visibilities is less trivial.

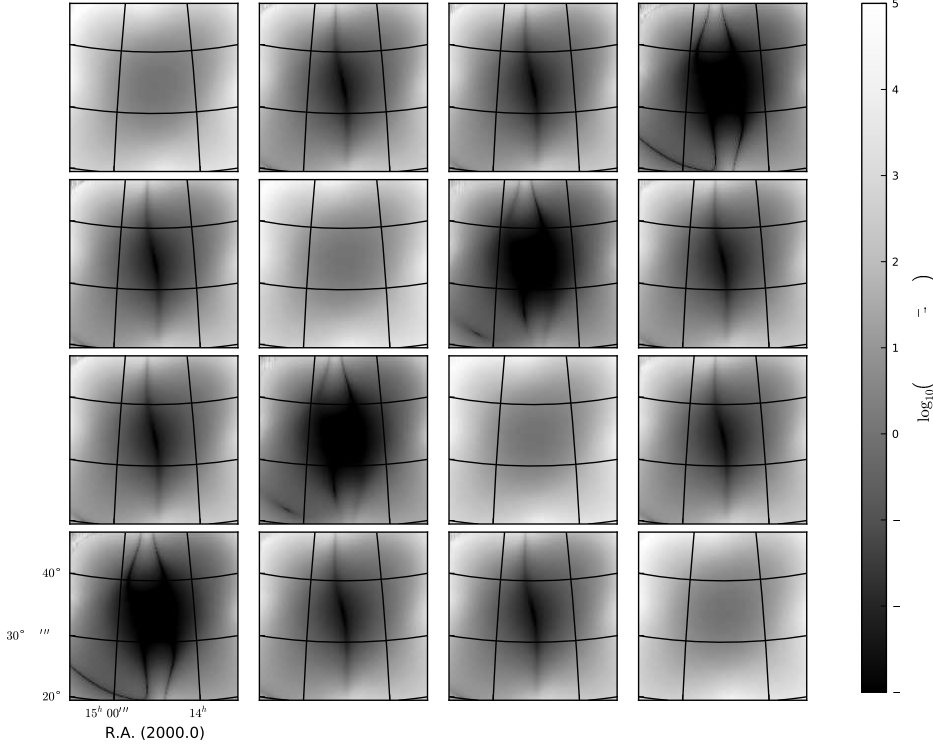


Fig. 2. This figure shows the correction $(\overline{\mathcal{D}_0^H \mathcal{D}_0})^{-1}$ that can be applied to the image before the minor cycle. This is a first order correction for the complicated phased array beam, that depends on time, frequency and baseline.

2.2. Estimating a sky image: polarization effects

There are different ways to solve for I from the set of visibilities, and reversing Eq. 3 relies on the linearity of the sky term in the measurement equation. As mentioned above our deconvolution scheme uses A-Projection. This generalization of CS-CLEAN is now better understood in the framework of compressed sensing, which include other new techniques (see McEwen & Wiaux 2011, for a review). Compressed sensing provides new ways to estimate the sky I for the set of visibilities. In this section we describe a method to approximately correct the polarisation effects in the image plane. In order to highlight the issues associated with polarization and baseline-dependence of the DDE, we simply write the sky least square solution \hat{I} as the pseudo inverse:

$$\hat{I} = (\mathcal{A}^H \mathcal{A})^{-1} \mathcal{A}^H V \quad (4)$$

where the term $\mathcal{A}^H V$ is the $4N_{pix}$ dirty image, and $(\mathcal{A}^H \mathcal{A})^{-1}$ is the $(4N_{pix}) \times (4N_{pix})$ image plane deconvolution matrix. Its structure is rather complicated and its size makes its estimate prohibitive.

In the simple case of a matrix \mathcal{D} being unity (no DDE), we can see that each 4×4 block number (x, y) of the matrix $\mathcal{A}^H \mathcal{A}$ is the instrumental response to a source centered at the location of the x^{th} pixel, evaluated at the y^{th} pixel. Therefore, computing $(\mathcal{A}^H \mathcal{A})^{-1}$ would involve estimat-

ing point spread function (PSF) centered at the location of each pixel and inverting a $4N_{pix} \times 4N_{pix}$ matrix. In the presence of non-trivial baseline-dependent DDE involving non-diagonal Mueller matrices, the problem becomes more complex. However, we show below that under some assumptions, the operator $(\mathcal{A}^H \mathcal{A})^{-1}$ (sometimes called the *deconvolution matrix*) affected by DDE is decomposable in a classical deconvolution matrix (containing information on the PSF), and a simple correction performed separately on each pixel.

Following the notation introduced above, we can write $\mathcal{A}^H \mathcal{A} = \mathcal{D}^H \mathcal{P} \mathcal{D}$ as a $4N_{pix} \times 4N_{pix}$ matrix, with $\mathcal{P} \equiv \mathcal{F}^H \mathcal{S}^H \mathcal{W}^H \mathcal{W} \mathcal{S} \mathcal{F}$ of size $4N_{blocks} N_{pix} \times 4N_{blocks} N_{pix}$. This later matrix is block diagonal, and each of its $4N_{pix} \times 4N_{pix}$ block describes the PSF of the instrument for a given baseline-time-frequency block. Their 4×4 xy -block are scalar matrices, the scalar $p_{t,\nu,pq}(x, y)$ being the response of the instrument evaluated at the y^{th} pixel to a source being centered at the x^{th} pixel in the given baseline-time-frequency block. We then have:

$$[\mathcal{A}^H \mathcal{A}](x, y) = \sum_{t,\nu,pq} p_{t,\nu,pq}(x, y) \mathcal{D}_{t,\nu,pq}^H(y) \mathcal{D}_{t,\nu,pq}(x) \quad (5)$$

It is virtually impossible to compute this matrix, and this illustrates the difficulty of doing an image plane deconvolution in the presence of time-frequency-baseline dependent DDE. In order to apply a first order correction

to the image plane, we assume that the direction dependent effects are constant enough across time, frequency, and baseline. Then we can write:

$$[\mathcal{A}^H \mathcal{A}](x, y) \sim N_{t,\nu,pq} p(x, y) \overline{\mathcal{D}^H \mathcal{D}}(x, y)$$

where $N_{t,\nu,pq}$ is the number of baseline-time-frequency blocks, $\overline{\mathcal{D}^H \mathcal{D}}(x, y)$ is a 4×4 matrix being the average of $\mathcal{D}_{pq}^{t,\nu}(y)^H \mathcal{D}_{pq}^{t,\nu}(x)$ over baselines, time, and frequency and $p(x, y)$ is the PSF stacked in baselines, time, and frequency. If the uv-coverage is good enough, then $\mathcal{A}^H \mathcal{A}$ is block diagonal ($p(x, y) = 0$ for $x \neq y$, $p(x, y) = 1$ otherwise). All the $x \neq y$ terms cancel out in the final product, and in the relevant part of $\overline{\mathcal{D}^H \mathcal{D}}$ matrix are the $N_{pix} 4 \times 4$ blocks on the diagonal. Applying $\overline{\mathcal{D}^H \mathcal{D}}^{-1}$ to $\mathcal{A}^H V$ can then be done on each pixel separately, by computing the product $\overline{\mathcal{D}^H \mathcal{D}}(x, x)^{-1} I_x$ where I_x contains the full polarization information (XX_x, XY_x, YX_x, YY_x) for the x^{th} pixel. This provides a way to estimate an approximate least square clean component value from a flux density in the dirty image in a given direction \mathbf{s}_x . The details of this image plane normalization are further discussed in Sec. 3.1. Although this normalization underlies a few assumptions, from the simulations presented in Sec. 4.3 we argue that the complex 4×4 matrix normalization per pixel presented here brings clear improvement. However it does not seem necessary for the sky solutions to convergence (see Sec. 7).

3. Implementation of A-Projection for LOFAR

As explained above, the full polarization A-Projection scheme has been described in Bhatnagar et al. (2008). However for the VLA implementation, due to the approximate unitarity of VLA beams, only the diagonal Mueller matrix terms had to be taken into account. As explained in Sec. 2, LOFAR has got very wide fields of view, and the baseline-dependent Mueller matrices associated to the 4-polarizations correlation products are non-diagonal. This basically means that each individual polarization cannot be treated independently from the others. In this section we describe in detail a few implementations of A-Projection allowing to correct for the non-diagonal Mueller matrices. We propose optimizations in relation to the telescope architecture. We will show in Sec. 4 that this algorithm can indeed deal with the heavily non-diagonal Mueller matrices associated with the LOFAR's wide fields of view.

Following Bhatnagar et al. (2008), we have build our implementation of A-Projection on top of a tradition Cotton & Schwab CLEAN algorithm (Schwab 1984). It is an iterative deconvolution algorithm that consists of two intricate steps. The CLEAN and A-Projection algorithms performs a series of operations that can be described in the following way:

$$\hat{I}^{n+1} = \hat{I}^n + \Phi \mathcal{A}^H (V - \mathcal{A} \hat{I}^n) \quad (6)$$

where the *true* sky estimate \hat{I}^{n+1} at step $n+1$ is built from its estimate \hat{I}^n at step n and Φ is a non-linear operator. It basically estimates the deconvolved sky from the residual dirty image $\mathcal{A}^H (V - \mathcal{A} \hat{I}^n)$. The minor cycle performs operations that approximate $(\mathcal{A}^H \mathcal{A})^{-1}$ discussed in Sec. 2, and takes the estimated image vector to zero but at the strongest components above a certain threshold. For the sky solutions to converge, the predict step ($\mathcal{A} \hat{I}^n$ or *degridding*) has to be accurate and unbiased. A-Projection is a fast way to apply \mathcal{A} or \mathcal{A}^H . If we separate the phase of the Fourier kernel in Eq. 1 as the sum of $\phi^0(u, v, \mathbf{s}) = u.l + v.m$ and $\phi^1(w, \mathbf{s}) = w.(\sqrt{1-l^2-m^2}-1)$, then following Bhatnagar et al. (2008) invoking the convolution theorem Eq. 1 becomes:

$$\begin{aligned} V_{pq}^{t,\nu}(u, v, w, i) &= \mathcal{F} \left(\sum_{j=1}^4 \mathcal{D}_{pq}^{t,\nu}(i, j, \mathbf{s}) W(w, \mathbf{s}) I_j(l, m) \right) \\ &= \sum_{j=1}^4 \left[\mathcal{F} (\mathcal{D}_{pq}^{t,\nu}(i, j, \mathbf{s}) W(w, \mathbf{s})) \right. \\ &\quad \left. * \mathcal{F} (I_j(l, m)) \right] \end{aligned} \quad (7)$$

where $*$ is the convolution product, \mathcal{F} is a 2D Fourier transform, $W(w, \mathbf{s}) = \exp(-2i\pi w.(\sqrt{1-l^2-m^2}-1))$, and i and j index the polarization number (running over (XX, XY, YX, YY)). This method is efficient because the DDE are *smooth* on the sky, meaning the support of the corresponding convolution function can be small (no high frequency terms). Indeed, as shown in Fig. 1 the LOFAR station beam is very smooth, and depending on the field of view the typical support is on the order of 5-11 pixels.

3.1. Naive implementation

The operation $\mathcal{A} \hat{I}$ in Eq. 6 converts a sky model into visibilities. To apply this operator on a massive amount of data in an algorithmically efficient way, we apply the scheme outlined in Eq. 7. First (i) a 2-dimensional fast Fourier transform is applied to the sky model image \hat{I} , on the 4 polarizations independently. Then (ii) for each baseline in a time-frequency block $\Delta(t, \nu)$ where the DDE is assumed to be constant, the 16 convolution functions are computed as the Fourier transform of the image plane Kronecker product of the DDE (for the LOFAR beam on a given baseline this block is typically 10 minutes and 0.2 MHz wide). The residual visibilities in each polarization are interpolated from the sum in Eq. 7. In practice, in order to minimize the support of the W-term and associated aliasing effect, we multiply the DDE in the image plane by a Prolate spheroidal function (see Appendix B for more details).

The predicted visibilities are removed from the measured visibility by computing the residual visibilities $V_{\text{residual}} = V - \mathcal{A} \hat{I}^N$. Applying \mathcal{A}^H applies a correction to the residual visibilities, and projects the result onto a grid

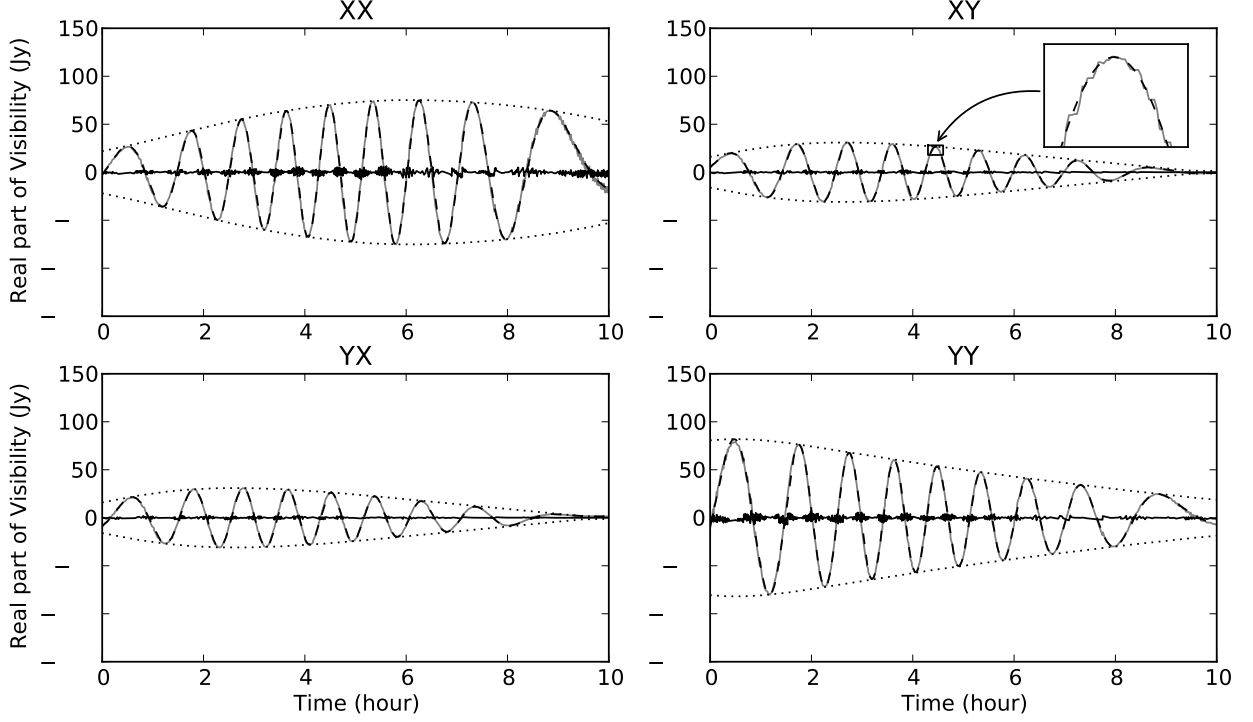


Fig. 3. The essential part of the A-Projection algorithm relies in the predict step, which transforms a 2D sky image (the projection of the celestial sphere on a plane) into a set of visibilities. We have simulated a dataset having one off axis source, where the *true* visibilities (black dashed line) have been estimated using Eq. 1 taking the beam into account. This plot shows the comparison in all measured polarizations between the exact value of the visibility of a given baseline, and the A-Projection estimate (gray line). Contrarily to a traditional predict step, the visibilities are modulated by the beam amplitude (dotted line) and we have time-dependent polarization leakage. The over-plotted graph shows a zoom in the small region shown on the top-right panel. In the degriding step, we use a computationally efficient closest neighbor interpolation, creating steps in the predicted visibilities.

(the *gridding* step) and Fourier transforms it. In practice this is done as follows:

$$V_{\text{corr},u,v,w,i}^{(t,\nu),(p,q)} = \sum_{j=1}^4 \left(\mathcal{F} \left(\mathcal{D}_{pq}^{*t,\nu}(j,i,s) W^*(w,s) \right) * V_{\text{residual},u,v,j}^{(t,\nu),(p,q)} \right) \quad (8)$$

and

$$I_{\text{dirty}}(l,m,i) = \mathcal{F}^{-1} \left(\sum_{(t,\nu),(p,q)} V_{\text{corr},u,v,w,i}^{(t,\nu),(p,q)} \right) \quad (9)$$

The resulting dirty image is still corrupted by the phased array beams related effects discussed in Sec. 2.2. Before the minor cycle we can either multiply each x^{th} 4-polarization $I_{\text{dirty}}(x)$ pixel by the 4×4 matrix $\overline{\mathcal{D}^H \mathcal{D}}(x)^{-1}$, or simply normalize each polarization of I_{dirty} by $\overline{\mathcal{D}^H \mathcal{D}}_{ii}(x)$, the diagonal elements of $\overline{\mathcal{D}^H \mathcal{D}}(x)$. As shown in Sec. 4.3, fully applying $\overline{\mathcal{D}^H \mathcal{D}}(x)^{-1}$ to each pixel in I_{dirty} shows a minor improvement.

The computational cost of taking DDE into account using A-Projection depends on (i) whether it is baseline dependent, (ii) the angular size at which the effect needs to be sampled (thereby constraining the size support of the convolution function) and (iii) the amplitude of the terms of the 4×4 $\mathcal{D}_{pq}^{t,\nu}(s_n)$ matrix. In the case of the full polarization A-Projection, the data needs to be corrected per baseline, per time and frequency slot. For each of those data chunk, in order to recover the corrected 4-polarization visibilities, one needs to take into account the 16 terms of the 4×4 $\mathcal{D}_{pq}^{t,\nu}(x)$ Mueller matrix, and the 4 visibilities built from the 2D Fourier transform of the sky model. Therefore in addition to the 16 convolution function estimate per baseline and time-frequency slot, in the gridding and degriding steps, one needs to compute $16 \times N_S^2$ operations per 4-polarization visibility point, where N_S is the support of the convolution function. The algorithmic complexity is further discussed in Sec. 4.4, but this implementation is too slow to be practical.

3.2. Separating antenna beam pattern and beam-former effect

Depending on the assumptions and system architecture one can find powerful algorithmic optimizations. We show here that in the case of LOFAR, we can use the fact that although stations are rotated one to another, the elementary antennas are parallel. The effective phased array beam $B_{p,s}$ of station p is modeled as $B_{p,s} = a_{p,s} \cdot E_{p,s}$, where $a_{p,s}$ is the array factor, and $E_{p,s}$ is the *element beam* pattern. The term $a_{p,s}$ depends on the phased array geometry and on the delays applied to the signal of the individual antennas before the summation (by the *beam-former* of each individual LOFAR stations). The term $E_{p,s}$ models both the individual element antenna sensitivity over the sky and its projection on the sky. We have

$$\text{Vec}(V_{pq}^{corr}) = \int_S \text{Vec}(a_{p,s} \cdot E_{p,s} \cdot I_s \cdot E_{q,s}^H \cdot a_{q,s}^*) \cdot \exp(-2i\pi\phi(u, v, w, s)) ds \quad (10)$$

with $a_{p,s}$ being scalar valued and $E_{p,s}$ is non-diagonal because (intuitively) the element beam projection on the sphere depends on the direction. Applying the convolution theorem to Eq. 10 we obtain:

$$\begin{aligned} \text{Vec}(V_{pq}^{corr}) &= \mathcal{F}[E_{q,s}^* \otimes E_{p,s}] \\ &\quad * \mathcal{F}[a_{p,s} \cdot a_{q,s}^* \cdot \exp(-2i\pi\phi^1(w, s))] \\ &\quad * \int_S I_s \cdot \exp(-2i\pi\phi^0(u, v, s)) ds \end{aligned} \quad (11)$$

All LOFAR stations have different layout on the ground, so the scalar valued array factor $a_{p,s}$ is station dependent. However all the individual antenna of all stations point at the same direction and we can assume that the Mueller matrix is baseline independent *ie* $E_{q,s}^* \otimes E_{p,s} = E_{0,s}^* \otimes E_{0,s}$. This requires an additional correction step of the gridded uv-plane visibilities but as we will see below, this is an interesting algorithmic shortcut because the element-beam correction can be applied on the baseline-stacked grids.

The element beam is very smooth over the sky and in most cases it can be assumed to be constant at time-scales of an hour, so that the polarization correction step does not need to be often applied. The degriding step goes as follows: (i) in each time-frequency slot $\Delta(t, \nu)_E$ where the Mueller matrix of the element beam is assumed to be constant, the polarization correction is applied to the (XX, XY, YX, YY) grids as the sum of convolved grids by the terms of $E_{0,s}^* \otimes E_{0,s}$. We then loop³ over baseline (pq), and time-frequency range $\Delta(t, \nu)_a$ where the array factor and w-coordinate are assumed to be constant within $\Delta(t, \nu)_E$. For each step in the loop (ii) the oversampled convolution function for baseline (pq) is estimated in $\Delta(t, \nu)_a$ for the term of the second line of Eq. 11, and (iii) it is used to interpolate the predicted visibilities at the given uv-coordinate, separately on each polarization.

³ We can parallelize the algorithm at this level.

As explained in Sec. 4.4, the computing time for estimating the convolution functions can be quite significant, and this scheme allows us to compute one convolution function per baseline instead of 16, and 4 gridding/degridding steps instead of 16. We note however that the assumption of baseline independence of the Mueller matrix on which is based this optimization starts to be wrong in the cases of direction dependent differential Faraday rotation, or for the longer baselines where the curvature of the earth starts to have an importance (in that case the element beam are not parallel). As discussed in Sec. 4.4, this computing time of this implementation is dominated by the convolution function estimate.

3.3. Separating the W-term: hybrid w-stacking

The support of the A-term is determined by the minimum angular scale to be sampled in the image plane. The beam or ionospheric effects are in general very smooth on the sky so that little amount of pixels are needed to fully describe the effects, therefore corresponding to a small convolution function support size (typically 11×11 pixels). The highest spatial frequency in the image plane is the W-term and its support can be as big as $\sim 500 \times 500$ pixel for the long baselines, wide fields of view, when the target field is at low elevation. This forces us to (i) compute a convolution function with a large support, and (ii) grid each individual baseline using the W-term dominated large convolution function.

We note however that the W-term is in itself baseline independent⁴: two baselines characterized with different ionosphere, beams, but with the same w-coordinate will have the same W-term. We therefore here slightly change the piping of the algorithm by taking into account the A-Term and the W-term separately as follows:

$$\begin{aligned} \text{Vec}(V_{pq}^{corr}) &= \mathcal{F}[E_s^* \otimes E_s] \\ &\quad * \mathcal{F}[\exp(-2i\pi\phi^1(w_{plane}, s))] \\ &\quad * \mathcal{F}[a_{p,s} \cdot a_{q,s}^* \cdot \exp(-2i\pi\phi^1(\Delta w, s))] \\ &\quad * \int_S I_s \cdot \exp(-2i\pi\phi^0(u, v, s)) ds \end{aligned} \quad (12)$$

We consecutively grid or degrid the data in w-slices *ie* that have similar w-coordinates. This algorithm is also known as W-stacking⁵. In addition, we take into account the fact that the points can lie above or below the associated w-plane central coordinate using the term $\exp(-2i\pi\phi^1(\Delta w, s))$, where $\Delta w = w - w_{plane}$. This step is similar to the traditional w-projection algorithm. If we have enough w-stacking planes, Δw is small, and the support of the baseline-time-frequency dependent convolu-

⁴ We talk about baseline dependence when a set of baselines with exactly the same uvw coordinates can give different visibilities.

⁵ See for example Maxim Voronkov's presentation at <http://www.astron.nl/calim2010/presentations> in the context of ASKAP

tion function remains small, leading to a dramatic decrease of the total convolution function estimation time. Conversely, given a convolution function support we can find the maximum usable Δw and derive the number of w-stacking planes as a function of the observation's maximum w coordinate (see Sec. 4.4.3 for more detailed discussion). In the case of LOFAR, choosing a convolution function support of ~ 21 pixel gives a number of w-stacking planes of ~ 30 .

This requires yet an additional step as compared to the implementation described in Sec. 3.2. We describe below the degriding step $\mathcal{A}\hat{I}^n$. First, following the notation introduced above (i) in each time-frequency interval $\Delta(t, \nu)_E$, we correct the 4-polarization grids from the element beam (including projection effects) using $E_{0,s}^* \otimes E_{0,s}$. Then (ii) we loop over the number of w-planes (ranging from $-w_{max}$ to w_{max} , see Sec. C), and convolve the grid obtained in (i) by the associated w-term (appearing in the second line of 12). Finally in (iii) for each w-plane obtained in each step of the loop (ii), we loop over the set of baselines $(pq)_w$ and time-frequency range $\Delta(t, \nu)_{a,w}$ associated with the given w-plane. We interpolate the predicted visibilities at the given uv-coordinate, separately on each polarization, based on the oversampled $\mathcal{F}[a_{p,s} \cdot a_{q,s}^* \cdot \exp(-2i\pi\phi^1(\Delta w, s))]$ convolution function. As discussed in Sec. 4.4.3, this is the fastest implementation of A-Projection we have obtained so far.

4. Simulations

In order to test the algorithmic framework described above we have performed a series of tests on LOFAR simulated datasets. In this section we summarize those results and discuss the computational costs of A-Projection for LOFAR.

4.1. One off-axis heavily polarized source

As discussed above by using A-Projection we can compute the *exact* value of the 4-polarization visibilities on a given baseline at a given time and frequency, from (i) the sky model and (ii) the baseline-time-frequency-direction dependent effects. In a first step, we focus on testing the degriding full polarization A-Projection degriding (or *predict*) step ($\mathcal{A}\hat{I}$). The accuracy of this step is indeed vital for the convergence of the CLEAN/A-Projection algorithm. Our experience suggest that any small numerical systematic bias in this operation can lead to strong divergence of CLEAN.

In order to test this transformation, we have simulated a dataset having only one polarized off-axis point-like source in a low-band 62 MHz LBA dataset. The 4-polarization visibilities have been generated using BBS (BlackBoard Selfcal⁶), which computes a direct calculation

of the visibilities following Eq. 1. It takes into account the beams of both the individual elementary antenna (and their projection on the sky), and the phasing of the individual antenna within a LOFAR station (the *array factor*). We located the simulated source a few degrees from the phase center and its flux density is non-physically polarized (with stokes parameters of I,Q,U,V=100,40,20,10 Jy). Fig. 3 shows the real part of BBS and A-Projection predicted visibilities on the baseline (01). The residuals are centered at zero, and A-Projection performs very well in predicting the 4-polarization visibilities, taking into account the complicated effects of the LOFAR phased array station's beams. A traditional predict using simple Fourier transform, facets or W-Projection would suffer from much higher residuals, driving systematics in the deconvolution, thereby limiting the dynamic range. Here the residual errors are dominated by the interpolation type we use in the uv domain (closest neighborhood, see B for details).

4.2. Dataset with many sources

4.2.1. LOFAR station beam

In order to test our modified implementation of the whole CLEAN algorithm (involving *gridding* and *degridding* steps), we have simulated a dataset containing 100 sources, with the source count slope following the 1.4 GHz NVSS source count (Condon et al. 1998). As in the dataset described above, the visibilities are generated using BBS⁶. We have taken into account the Jones matrices of the individual elementary antenna as well as the array-factor (the beam-forming step). As explained in Sec. 3.2 and shown in Fig. 3, as the LOFAR stations are rotated one to another, all baselines will be effected by beam effects differently. We have corrected the visibilities for the beam effect at the first order by computing $V_{pq,corr}^{t,\nu} = [D_p^{t,\nu}]_{s_0}^{-1} V_{pq}^{t,\nu} [D_q^{t,\nu,H}]_{s_0}^{-1}$, where $D_p(s_0)$ is the Jones matrix of station p computed at the center of the field. This mostly compensates for the element beam effects, and more specifically the projection of the dipoles on the sky. However, as shown below, the LOFAR fields are big, and the projection of the dipoles vary across the field of view. Most of the sources thereby generated have flux densities comprised between 10^{-2} and 1 Jy. We have added two bright sources of 100 and 10 Jy.

Fig. 4 shows the restored images produced using three different modified CLEAN algorithm. Each of those image contains ~ 3000 pixel in side, and is ~ 6 degrees wide. We have used 15 w-stacking, 128 Δw -planes (see Sec. 3.3), a maximum baseline of 5 k λ , a Briggs⁷ weighting scheme, a cycle-factor of 2.5 and a number of minor cycle iterations of 10000. The first map has been generated using W-Projection (Cornwell et al. 2008) as implemented in CASA⁸. Strong artifacts are present around the bright-

⁶ See [http://www.lofar.org/wiki/doku.php?id=public:documents:lofar_documents&s\[\]=bbs](http://www.lofar.org/wiki/doku.php?id=public:documents:lofar_documents&s[]=bbs) for a review of BBS functionalities

⁷ With a robust parameter of 0, see <http://www.aoc.nrao.edu/dissertations/dbriggs>

⁸ <http://casa.nrao.edu>

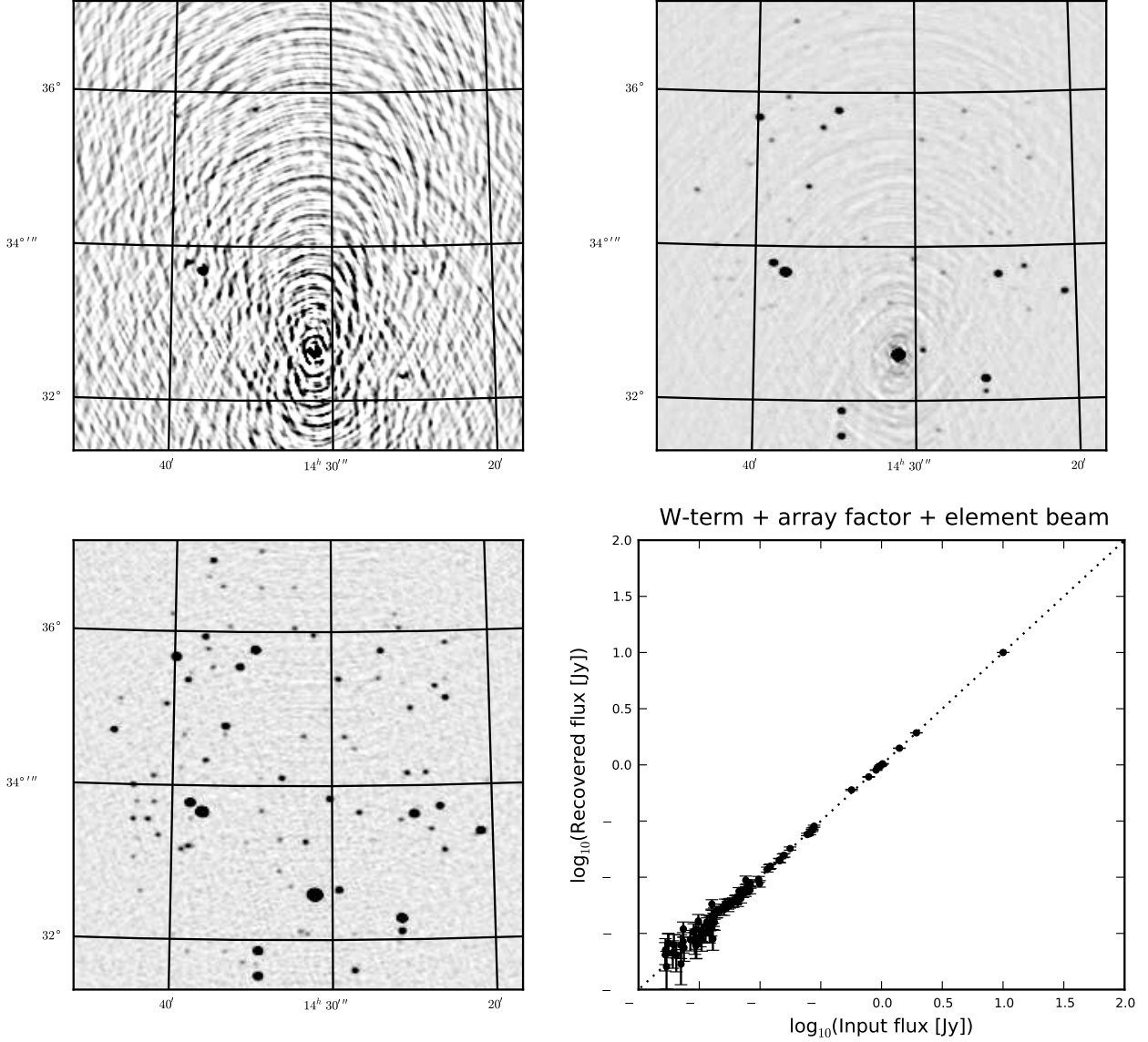


Fig. 4. This figure shows the dramatic effect of the LOFAR phased array beams for a simulated dataset. Specifically, the visibilities have been generated taking into account (i) the individual antennas, (ii) their projection on the sky and (iii) the beam-forming step (the scalar *array factor*). The top-left image shows the deconvolved sky as estimated with a traditional imager not taking into account time-frequency direction dependent effects. The top-right and bottom-left have been generated by taking into account the array factor only and both array factor and the element beam respectively. The bottom right panel shows the input flux densities are correctly recovered.

est off-axis source, and the dynamic range reaches 1 : 230. In the second image we have used our implementation of A-Projection taking into account the array factor only. Taking that effect into account the effect of the lower the residual visibility levels on each individual baselines and the dynamic range reaches $\sim 1 : 3.400$. In the third image

we have taken into account all the LOFAR beam effects: the individual antenna sensitivity, their spatially varying projection, and the array factor. The dynamic range increases to $\sim 1 : 12.000$ ⁹

⁹ It seems the output images of our imager is presently limited at $\sim 10^{-4}$ accuracy for some numerical precision prob-

The sources flux densities in the restored maps have been extracted using the LOFAR source extraction software pyBDSM¹⁰. As shown in Fig. 4, the input flux densities are very well recovered.

4.2.2. LOFAR station beam and ionosphere

In order to test the ionospheric correction with A-Projection, we have simulated a dataset containing 100 sources, affected by a simulated ionosphere. The ionospheric effects essentially consist of a purely scalar, direction dependent phase. Faraday rotation is not included. In addition to this purely scalar ionospheric phase effect the visibilities are affected by the direction dependent LOFAR's stations beam effects discussed above.

The ionosphere is modeled as an infinitesimally thin layer at a height of 200 km. The Total Electron Content (TEC) values at a set of sample points are generated by a vector autoregressive (var) random process. As described in van der Tol (2009) the spatial correlation is given by Kolmogorov turbulence. The values at intermediate points are found using Kriging interpolation. The set of sample points are the pierce points for five points in the image located at the four corners and the centre. This way the ionospheric layer is sampled in the most relevant area. There are at least five sample points within the field of view of each station.

Fig. 5 shows the dirty image at the location of a given source before and after A-Projection correction of beam and ionosphere. This suggests that the dirty undistorted sky is properly recovered from the corrupted visibilities. We compare in Fig. 6 the cleaned image with and without the A-Projection correction. Those simulations demonstrate that A-Projection can indeed deal with the complicated effects associated with the ionosphere.

4.3. Convergence

We have also studied the influence of the various corrections described in this paper on the convergence speed of the estimated flux densities through the major cycle loop. For this we have used the dataset described in Sec. 4.1 containing one off-axis source having stokes (IQUV)=(100,40,20,10) Jy.

Fig. 7 shows the evolution of the estimated flux density as a function of the major cycle iteration number for different algorithm. In the first panel, the W-Projection algorithm somewhat converges to the *observed* flux density

lem: all the code uses single precision floating point numbers, and the roundings of products and sums involved in the algorithm seems to generate limitations at this level. Detailed tests based on multi-threading and single threading comparison consistently reveal an instability at this level. For the LOFAR surveys however, we might not need higher accuracy, as we plan to use direction dependent peeling to substract the brightest sources on the first $\sim 1:10^2$ dynamic range.

¹⁰ See <http://www.lofar.org/wiki/doku.php> for more information.

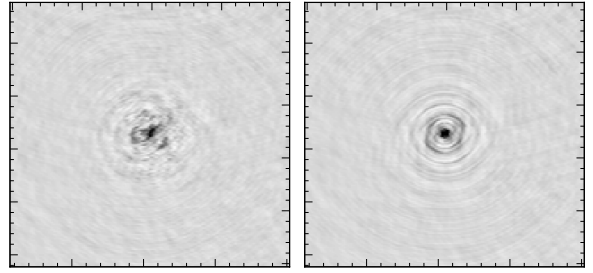


Fig. 5. This figure show the dirty image at the location of a bright source in a simulated dataset. The images are 1 deg in diameter. In the image of the left panel the dirty image shows important distortion as a result of ionospheric effect. The A-Projection correction (right panel) for an ionospheric phase screen clearly shows improvement.

(to the "beam-multiplied" sky). The situation is better using full polarization A-Projection without the element beam (therefore assuming the dipole projection on the sky are constant across the field of view). We note that in the absence of polarization, the situation is not as bad. Taking the element beam into account, the algorithm makes the estimated flux densities to converge to the true values to better than a percent. In this version of the algorithm, the image plane correction is the same on all polarization and is just a scalar normalization (the average primary beam normalization). The situation gets slightly better in terms of convergence speed by applying the image-plane renormalization described in Sec. 2.2.

In any case the accuracy of the recovered flux densities seems to be guaranteed by the accuracy of the degriding step. Our experience in implementing A-Projection suggest that any small systematic error in the degriding step can lead to biases in the recovered flux densities or divergence in the more severe cases. The image-plane polarization correction, seems to be bringing something positive to the convergence speed, but that step appears not to be necessary.

4.4. Computational and memory-related costs

Given the large amount of data that have to be processed in the imaging of an interferometric dataset, reducing the algorithmic complexity is of primary importance.

4.4.1. Memory-related issues

The A-term is generally very smooth in the image plane, with corresponding small support convolution functions and under those conditions the A-Projection algorithm is virtually free as explained in Bhatnagar et al. (2008) (and in Cornwell et al. 2008, in the case of W-Projection): the A-term and the low w-coordinate convolution function support is less or comparable to the spheroidal func-

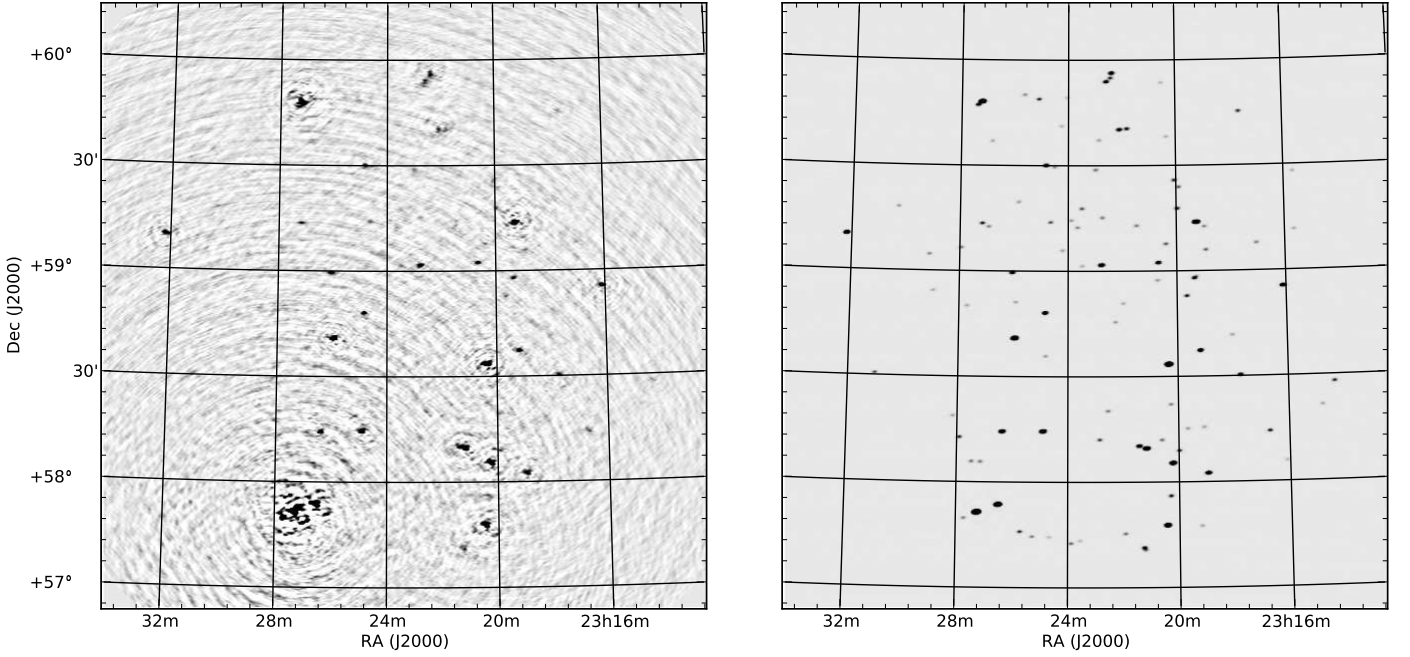


Fig. 6. This figure shows the deconvolved image synthesized from the simulated dataset described in Sec. 4.2.2. In the left panel, the ionospheric effects are not taken into account, and our deconvolution scheme naturally produces severe artifacts and high residuals in the reconstructed sky. The deconvolved image shown in the right panel has been estimated using our implementation of A-Projection with the time-dependent ionospheric phase screen.

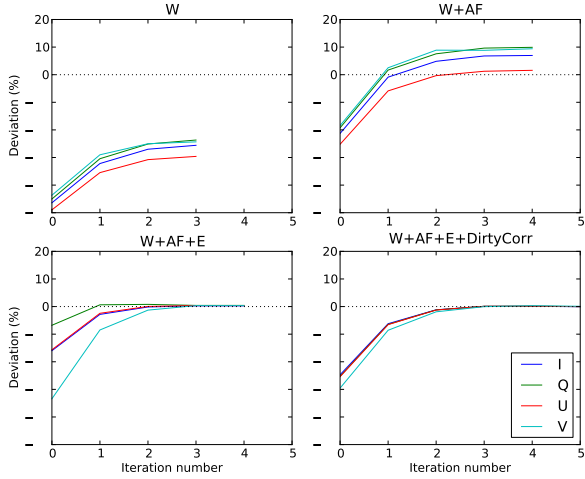


Fig. 7. This figure shows the evolution of the estimated flux density as a function of the major cycle iteration for one polarized off-axis source. From top to bottom, left to right are generated (i) using W-Projection only, (ii) using the AW-Projection with the array factor only, (iii) using AW-Projection taking the full beam model into account. Finally in (iv) we show the effect of doing the image plane correction described in Sec. 2.2.

tion support that anyway needs to be applied in a traditional gridded/degridded. However, as described in Sec. 3.2, all LOFAR stations are rotated one to another, and the synthesized stations beams are all different in a given direction. This gives rise to a serious algorithmic complication because the convolution functions become baseline-dependent: the number of possible convolution functions is $16 \times N_{\text{times}} \times N_{\text{Freqs}} \times N_{\text{Stations}} \times (N_{\text{Stations}} - 1)/2$. With ~ 800 to ~ 1500 baselines, even with a convolution function every 10 minutes, a typical observing run of 8 hours, one frequency channel, and an average support size of ~ 30 pixel (taking into account the w-term), this gives a ~ 100 Gbytes of needed storage. This is optimistic as in the case of other DDE such as the ionosphere, the A-term will have to be evaluated every 30 seconds. Even if the storage is done at the minimal necessary resolution, those numbers are clearly prohibitive for storing the convolution into memory. The convolution functions therefore have to be computed on the fly, and algorithmically, this represents an additional cost.

4.4.2. Naive and element-beam-separated A-Projection

The LOFAR station's beams are smooth on the sky, and the corresponding convolution function support are small (typically 11-15 pixel complex image), while the W-term needs up to $\gtrsim 500$ pixel depending on the w-coordinate (with an average of ~ 30 pixels). The computing time de-

depends on the convolution function support N_S , and for the implementations described in Sec. 3.1 and 3.2, we have:

$$N_S = \sum_{cf=\{S, A_p, A_q, W\}} N_S(cf) \quad (13)$$

where the subscripts S , A_p , A_q , and W stand for the Prolate spheroidal ($\sim 7 - 11$ pixels), A-terms of antenna p and q and W-term. For a typical field of view, we have $N_S(A) = 9 - 15$ and $N_S(W) \propto D^2.w$, where D is the field of view diameter and w is the W-coordinate of the given baseline in the given time-slot (see Appendix B). For that baseline, time and frequency slot we can write the total computing time as $t_{pq,tot}^{tv} = t_{pq,grid}^{tv} + t_{pq,CF}^{tv}$ where t_{grid}^{tv} and t_{CF}^{tv} are the gridding and the convolution function estimate times. In most cases for LOFAR data, we have $N_S \sim N_S(W)$ and $t_{grid} \propto D^4 w^2$. Our experience has shown that t_{CF} is dominated by the computing time of the Fourier transform of the zero-padded convolution function (see Fig. 8), which size is $\mathcal{O}.N_S$ where \mathcal{O} is the oversampling parameter which controls the quality of the nearest neighborhood interpolation. If $N_{buf}^W \propto 1/(\Delta T_{win} \Delta \nu_{win})$ is the number of visibility buffers associated with the W-plane (ΔT_{win} and $\Delta \nu_{win}$ are the time/frequency window interval in which the DDE are assumed to be constant), then we have $t_{CF}^W \propto N_{buf}^W . \mathcal{O}^2 w^2 D^4 . \log(\mathcal{O} w D^2)$. The gridding time for a given w-plane is simply $t_{grid}^W \propto N_{vis}^W w^2 D^4$ where N_{vis}^W is the number of visibilities associated with the w-plane. The total computing time can be written as:

$$t_{tot} \sim N_{el} t_{el} + \left(\sum_{W\text{-planes}} \left[a t_{grid}^W + b t_{CF}^W \right] \right) \quad (14)$$

where a and b are constants, N_{el} is the number of time/frequency blocks in which the element-beam is assumed to be constant, and $t_{el} \propto c N_{pix}^2 (1 + 2 \log(N_{pix}))$ is the computing time necessary to apply the element-beam to the grids ($c = 16$ for full polarisation imaging, $c = 8$ for I-stokes only). For the implementation described in Sec. 3.1, we have $t_{el} = 0$, but both a and b are 16 times higher (8 for I-stokes only) than in the case of the algorithm described in Sec. 3.2. Fig. 8 shows the gridding and convolution function estimate times as a function of the w-coordinate of the given baseline in the given time-slots. From that figure it is clear that (i) the W-term is the most important limiting factor and (ii) the estimate of the convolution function represents a major limitation of those implementations, especially in the cases of a quickly varying DDE such as the ionosphere where the convolution function calculation would largely dominate.

4.4.3. Hybrid W-Stacking and A-Projection

As explained in Sec. 3.3 for the W-staking implementation, the baseline-time-frequency-dependent oversampled convolution function is the Fourier transform of the zero-padded image-plane product of the spheroidal, the A-term

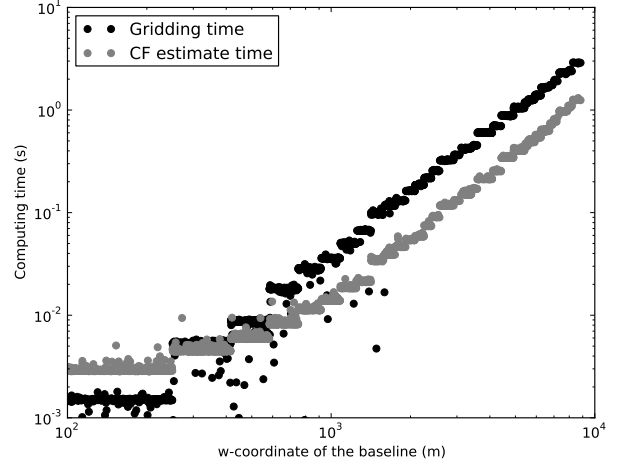


Fig. 8. Due to their different orientation, all LOFAR station have a different synthesized primary beam. The corresponding convolution function for applying the A-projection are therefore baseline-dependent, and we have to compute them on the fly. This figure is showing the computing time of both the gridding and convolution function estimate time for a 10 degree field of view, a maximum w-coordinate to be 10^4 meters, and a time-window of $T_{window} = 1200$ seconds. When the DDE quickly varies, a convolution function is often required, and the convolution function computing time can largely exceed the gridding computing time.

and a W-term accounting for the Δw -distance between the given visibilities and the corresponding w-plane. The bigger is the support of the convolution function and the less w-planes we need to fully correct for the w-term. As explained in Appendix B, in order to properly sample the w-term in the image plane, to a given w-coordinate and field of view correspond a convolution function support. We can then obtain $\Delta w = a.N_S/D^2$ (with $a = \sqrt{2}/(4\pi)$), and the needed number of w-planes between $-w_{max}$ and w_{max} is $N_W = w_{max} D^2 / (a.N_S)$. We compute the W-term convolution in the image plane so this step cost goes as $t_W \propto N_{pix}^2 (1 + 2 \log(N_{pix}))$. The total computing time is then:

$$t_{tot} \sim b N_{vis} N_S^2 + c N_{buf} \mathcal{O}^2 N_S^2 \log(\mathcal{O} N_S) + N_{el} [t_{el} + N_W t_W] \quad (15)$$

where N_{el} is the number of time/frequency buffers in which the element-beam is assumed to be constant, N_{buf} is the number of time/frequency buffer where the A-/ Δw -term are assumed to be constant, b and c are constants. In Tab. 1 we present the typical computing times for a major cycle with the implementation discussed here and presented in Sec. 3.3.

Table 1. Computational performance of our fastest A-Projection implementation described in Sec. 3.3 and 4.4.3. Columns display the various performance corresponding to different imaging settings. Those tests were performed for a 12 subbands dataset (1 channel per subband), on the CEP2 LOFAR cluster node each havin 24 AMD Opteron(tm) 6172 Processors, and 64 Gb of RAM memory. t_{tot}^{grid} is the total for a gridding or degriding step. The times t_{CF} , t_{grid} , t_{el} , t_W are given in fraction of t_{tot}^{grid} , and correspond to the times needed to compute convolution functions, to gridding/degriding the data, to apply the element beam, and to apply the W-term respectively. We are confident we can still win a factor of $\gtrsim 2 - 4$ with respect to those times values.

N_{pix}	d (")	D (deg)	N_{stokes}	w_{max} (km)	N_S	N_W	Δ_t, Δ_ν (s, MHz)	$\Delta_t^{el}, \Delta_\nu^{el}$ (h, MHz)	t_{CF} (%)	t_{grid} (%)	t_{el} (%)	t_W (%)	Memory (Gb)	t_{tot}^{grid} (sec.)
1024	40	11.4	IQUV	20	15	14	300, 0.78	3, 3.5	67.8	10.2	19.7	2.1	1	40.9
4096	10	11.4	IQUV	20	15	14	300, 0.78	3, 3.5	24.5	4.9	57.5	12.8	4.5	117.4
4096	10	11.4	I	20	15	14	300, 0.78	none	52.0	11.4	0.0	36.5	1.5	143.7
8192	5	11.4	I	20	15	14	300, 0.78	none	32.8	6.1	0.0	60.9	6.1	335.7

5. Summary and future work

The new generation of low-frequency interferometers (EVLA, MWA, LWA) and precursors of the SKA (LOFAR, ASKAP, MeerKAT) cannot be properly used without the development of new techniques to calibrate and apply in the imaging step the many direction dependent effects influencing the electro-magnetic field. These effects mainly include the antenna/stations complicated beam effects, ionosphere phase and amplitude fast variation and associated Faraday rotation.

In this paper we have mainly discussed the issues associated with the application of the LOFAR elementary antenna and station beams, which involve the usage of a few levels of phase arrays. Using the Measurement equation formalism, the associated high complexity (wide field of view, individual station rotations, projection of the elementary dipoles on the sky), the problem of imaging and deconvolution of LOFAR calibrated dataset can be solved by applying the A-Projection algorithm described in Bhatnagar et al. (2008). Due to its very large field of view ($\sim 5 - 10$ degrees in diameter), a full-polarization A-Projection implementation dealing with non-diagonal Mueller matrices is needed for LOFAR. In this paper we have shown that A-Projection can indeed deal with the heavily baseline-dependent non-diagonal direction-dependent Mueller matrices associated with LOFAR baselines. We have also demonstrated that efficient ionospheric correction can be performed using A-Projection. We have proposed a series of implementations of A-Projection for LOFAR taking into account non-diagonal Mueller matrices, aiming at accuracy and computing efficiency.

5.1. Optimizations

As explained in Sec. 4.4 the DDE although varying quickly are smooth on the sky, so the convolution functions have a small support. However, in the case of LOFAR the wide field of view imposes to take into account (i) the W-term, and (ii) the off-diagonal terms of the Mueller matrices (due to the varying projection of the dipoles on the sky,

or to the Faraday rotation). In addition (iii) the baseline dependence make the storage of the convolution functions to be prohibitive and those have to be computed on the fly.

In our first implementation (Sec. 3.1) the constraint (i) makes the W-term convolution function support to often dominate, while (ii) requires to set up a complicated machinery in the gridding and degriding step, taking into account all polarizations to correct for polarization leakage. In the case of a quickly varying DDE such as the ionosphere, the step (iii) can completely dominate the computing time (usually set by the gridding/degriding times).

Using the fact that LOFAR station's elementary antennas (responsible for the complicated projection effects) are parallel although the station's layout are rotated, we could separate the first implementation in two steps (Sec. 3.2). The first is a purely scalar gridding/degriding and suffers from (i) and (iii), while the second is only affected by (ii). The non-diagonality of the Mueller matrix is corrected on the baseline-stacked uv-plane, so we win a factor between 10 and 16 as compared to the first implementation. It is important to note that this optimization breaks down when the non-diagonal Mueller matrix becomes baseline-dependent such as in the cases of very long baselines (due to the earth's curvature), or to the ionosphere's differential Faraday rotation.

In the last implementation (Sec. 3.3), we still apply the direction-dependent non-diagonal Mueller matrix of the baseline-independent element-beam, and go further by separating the W-term from the A-Term. The former is responsible for the large support sizes and is baseline independent (around a given w-plane), while the later has small support and is baseline dependent. In this implementation, we grid/degrid the data based on a small support baseline-dependent oversampled convolution function, and convolve the input or the output grids (forward and backward steps respectively) with the non-oversampled W-term convolution function. We save computing time by computing the oversampled convolution function of a generally much smaller support, and the net

gain lies between 2 and 10 as compared to the seconds implementation (Sec. 3.2, and Tab. 1).

Such optimizations are vital for the feasibility of the LOFAR extragalactic surveys, given their huge integration times totalizing hundreds of days. For the SKA, it will be important to take the algorithmic optimization into account, as they are linked to the instrument and system architecture. They can reduce the algorithmic complexity by orders of magnitude.

5.2. Real LOFAR data and future work

We have conducted many experiments with LOFAR in order to test the our imager on real LOFAR data. Specifically, we have observed the same set of objects, in different observations having their pointing centers shifted by few degrees. Compared to a traditional imager (CASA), our implementation of A-Projection gives coherent corrected flux densities at the level of 5-10% on the edge of the field. However, although we have shown in this paper that the algorithm is giving excellent results on simulated dataset, it shows little or no dynamic range improvement in the resulting images, as compared to a traditional imager.

LOFAR is however a very complex machine and feeding the imager with a wrong direction-dependent calibration solution will not lead to any improvement in the deconvolved sky, and can of course even decrease the dynamical range. In order to improve LOFAR's dynamic range we will have to make progress in understanding the calibration aspects of DDE, especially those related to ionosphere and differential Faraday rotation. Much effort is spent on that direction, and DDE calibration algorithms of low complexity are under development or have already been achieved such as SAGECal (Yatawatta 2008).

On the imager side, further ongoing development include (i) implementation on GPU, of either or both of the gridding/degridding or convolution function calculation, (ii) compressed sensing in the image plane, (iii) uv-plane interpolation techniques different from zero-padding FFT, and (iv) Wide-Band A-Projection (Bhatnagar et al. 2012). Ionosphere and true beam calibration, in combination with peeling techniques, will hopefully allow us to use the framework presented in this paper to reach the high $\sim 10^5 - 10^6$ dynamic range needed to construct the deepest extragalactic LOFAR surveys, .

Acknowledgements. This work was done using the R&D branch of the CASA code base. We would like to thank the CASA Group and casacore team for the underlying libraries. We thank Sarod Yatawatta for the useful discussions on the element beam modeling that lead to powerful optimizations.

References

Bernardi, G., Mitchell, D. A., Ord, S. M., et al. 2011, MNRAS, 413, 411
 Bhatnagar, S. 2009, in Astronomical Society of the Pacific Conference Series, Vol. 407, The Low-Frequency Radio

Universe, ed. D. J. Saikia, D. A. Green, Y. Gupta, & T. Venturi, 375–+
 Bhatnagar, S., Cornwell, T. J., Golap, K., & Uson, J. M. 2008, A&A, 487, 419
 Bhatnagar, S., Rao, U., & Golap, K. 2012, A&A
 Condon, J. J., Cotton, W. D., Greisen, E. W., et al. 1998, AJ, 115, 1693
 Cornwell, T. J., Golap, K., & Bhatnagar, S. 2008, IEEE Journal of Selected Topics in Signal Processing, 2, 647
 de Vos, M., Gunst, A. W., & Nijboer, R. 2009, IEEE Proceedings, 97, 1431
 Hamaker, J. P., Bregman, J. D., & Sault, R. J. 1996, A&AS, 117, 137
 Heald, G., McKean, J., Pizzo, R., et al. 2010, in ISKAF2010 Science Meeting
 McEwen, J. D. & Wiaux, Y. 2011, ArXiv e-prints
 Mitchell, D. A., Wayth, R. B., Bernardi, G., Greenhill, L. J., & Ord, S. M. 2012, Journal of Astronomical Instrumentation, 1, 50003
 Rau, U. & Cornwell, T. J. 2011, A&A, 532, A71
 Schwab, F. R. 1984, AJ, 89, 1076
 Smirnov, O. M. 2011, A&A, 527, A106
 Sullivan, I. S., Morales, M. F., Hazelton, B. J., et al. 2012, ApJ, 759, 17
 van der Tol, S. 2009, PhD thesis, TU Delft
 Yatawatta, S. 2008, ArXiv e-prints

Appendix A: Measurement Equation formalism

In order to model the complex direction dependent effects (DDE - station beams, ionosphere, Faraday rotation, etc), we make extensive use of the Measurement Equation formalism developed by Hamaker et al. (1996). The Measurement Equation provides a model of a generic interferometer. Each of the physical phenomena that transform or convert the electric field before the correlation computed by the correlator are modeled by linear transformations (2×2 matrix). If $\mathbf{s} = (l, m, n = \sqrt{1-l^2-m^2})$ is a sky direction, and H stands for the Hermitian transpose operator, then the correlation matrix V_{pq} between antennas p and q , can be written as follows:

$$V_{pq}^{meas} = G_p \left(\sum_{\mathbf{s}} D_{p,\mathbf{s}} \cdot K_{p,\mathbf{s}} \cdot F_{\mathbf{s}} \cdot F_{\mathbf{s}}^H \cdot K_{q,\mathbf{s}}^H \cdot D_{q,\mathbf{s}}^H \right) G_q^H \quad (\text{A.1})$$

where $D_{p,\mathbf{s}}$ is the product of direction-dependent Jones matrices corresponding to antenna p (e.g., beam, ionosphere phase screen, and Faraday rotation). G_p is the product of direction-independent Jones matrices for antenna p (like electronic gain and clock errors). The matrix $K_{p,\mathbf{s}}$ describes the effect of the array geometry and correlator on the observed phase shift of a coherent wavefront between antenna p and a reference antenna. This effect is purely scalar so it is reducible to the product of a scalar and the unity matrix, so we can write $K_{p,\mathbf{s}} \cdot K_{q,\mathbf{s}}^H = \exp(-2i\pi\phi(u, v, w, \mathbf{s})) \cdot \mathbf{1}$ where (u, v, w) is

the baseline vector between antenna p and q in wavelength units and $\mathbf{1}$ is the unity matrix. We then have $\phi(u, v, w, \mathbf{s}) = u.l + v.m + w.(\sqrt{1-l^2-m^2} - 1)$, where the -1 term models the correlator effect when phasing the signals in the direction of w . Finally the product $F_{\mathbf{s}} \cdot F_{\mathbf{s}}^H$ is the sky contribution in the direction \mathbf{s} and is the true underlying source coherency matrix $[[XX, XY], [YX, YY]]$.

This elegant formalism enables us to model the full polarization of the visibility as a function of the *true* underlying electric field correlation. In a simple and consistent way it takes the direction dependent and direction independent effects into account. Indeed most of the Jones matrices in the measurement equations have a fairly simple formulation and radio calibration problem amounts to finding the various components of G and E , given a sky model $I_{\mathbf{s}} = F_{\mathbf{s}} \cdot F_{\mathbf{s}}^H$.

The measurement equation introduced above can be written in a more extended and continuous form better suited for imaging by applying the Vec operator¹¹ to V_{pq}^{corr} . We obtain:

$$\text{Vec}(V_{pq}^{corr}) = \int_S (D_{q,s}^* \otimes D_{p,s}) \cdot \text{Vec}(I_{\mathbf{s}}) \cdot \exp(-2i\pi\phi_{pq}(u, v, w, \mathbf{s})) ds \quad (\text{A.2})$$

where \otimes is the Kronecker product, Vec is the operator that transforms a 2×2 matrix into a dimension 4 vector.

Appendix B: Further algorithm details

In this section we describe some important details of the various implementations of A-Projection for LOFAR. In particular, we make extensive use of the Prolate spheroidal function for resolution adaptation and zero-padding for uv-plane interpolation.

As explained in Sec. 3, since LOFAR stations are characterized by different primary beams, the gridding and de-gridding steps are baseline dependent. Therefore the convolution functions cannot be computed once and kept in memory as is done for W-projection. Instead, they have to be computed on the fly (see Sec. 4.4). However, because the station's beam model is fairly complex and costly to evaluate (coordinates transformation, estimate of the Element beam Jones matrix), we store in memory at the minimal resolution the 4-polarization image plane beams (their Jones matrices). The necessary resolution is simply estimated as $\lambda/(2.D_{station})$, where $D_{station}$ is the given station's diameter.

The W-term is also estimated once and stored in memory at the minimal resolution. This amounts to finding the maximum frequency to be sampled in the image plane, and the corresponding number of pixels corresponding to the minimum support required for the W-term convolution function. If the image is of angular diameter D_{im} ,

the necessary resolution needed to properly sample to W-term is the inverse of the highest spatial frequency, located in one of its corners. The support of the W-term is then $N_W = (4\pi w D_{im}^2) / \sqrt{2 - D_{im}^2} \sim 4\pi w D_{im}^2 / \sqrt{2}$.

In order to interpolate the visibilities on the grid in the gridding step (or conversely in the de-gridding step), or to adapt the resolution of the A- and W-terms we use a zero-padding interpolation. This scheme can produce artifacts due to the presence of sharp edges and aliasing problems, so we have to make extensive use of a Prolate spheroidal function. It is computed at the maximum resolution in the image plane. We then Fourier transform it, find its support $N_S(S_p^{uv})$, "cut" it to that size, and store it in the uv-plane (hereafter S_p^{uv}).

For the various algorithms presented in this paper, we have to compute the products of various DDE in the image plane. For example for the algorithm described in Sec. 3.1, we have to adapt the A- and W-terms resolution before multiplying them in the image plane. We first have to find the support N_S of the convolution function as in Eq. 13. We first compute the image plane spheroidal at the resolutions of the A- and W-terms ($S_p^{N_S(A)}$ and $S_p^{N_S(W)}$ respectively) as follows:

$$\begin{aligned} S_p^{N_S(A)} &= \mathcal{F}^{-1} \mathcal{Z}^{N_S(A)} S_p^{uv} \\ S_p^{N_S(W)} &= \mathcal{F}^{-1} \mathcal{Z}^{N_S(W)} S_p^{uv} \end{aligned} \quad (\text{B.1})$$

where \mathcal{F} is the Fourier transform, \mathcal{Z}^n is the zero-padding operator that puts the input into a grid of the size n . To estimate the A-term interpolated on an $N_S \times N_S$ pixel image:

$$A^{N_S} = (S^{N_S})^{-1} \mathcal{F}^{-1} \left(\mathcal{Z}^{N_S} \mathcal{F} \left(S_p^{N_S(A)} A \right) \right) \quad (\text{B.2})$$

We obtain the image plane effects at the same resolution and multiply them according to the specific needs of the various implementations described in Sec. 3 and obtain the image plane product P_{im} . We then compute the oversampled convolution function as:

$$CF^{\mathcal{O}N_S} = \mathcal{F} \left(\mathcal{Z}^{\mathcal{O}N_S} (S^{N_S} P_{im}) \right) \quad (\text{B.3})$$

where the resulting interpolated convolution function $CF^{\mathcal{O}N_S}$ has $\mathcal{O}N_S \times \mathcal{O}N_S$ pixels with \mathcal{O} the oversampling parameter. If the final image is of size $N_{pix} \times N_{pix}$, because we have used the spheroidal function, after applying the inverse Fourier transform, we have to normalise the dirty image by the spheroidal function $S_p^{N_{pix}} = \mathcal{F}^{-1} \mathcal{Z}^{N_{pix}} S_p^{uv}$.

As outlined above, all LOFAR stations are different, and the convolution functions are baseline dependent. The *for* loops described in Sec. 3 are therefore parallelized on baseline. For the de-gridding step from a common read-only grid, the residual visibilities are estimated independently using different threads. The code has been created in the LOFAR package and is dependent on the casacore and casarest packages.

¹¹ The Vec operator transforms a matrix into a vector formed from the matrix columns being put on top of each other. It has the following useful properties: (i) $\text{Vec}(\lambda A) = \lambda \text{Vec}(A)$, (ii) $\text{Vec}(A + B) = \text{Vec}(A) + \text{Vec}(B)$, and (iii) $\text{Vec}(ABC) = (C^T \otimes A) \cdot \text{Vec}(B)$.

Appendix C: Limiting the maximum w-coordinate

For the traditional interferometers at high frequencies, in general the field of view is small enough so that the W-term can always be neglected ($w \cdot (\sqrt{1 - l^2 - m^2} - 1) \sim 0$). However, for the wide fields of view, long baselines interferometers, the W-term is of great importance, and not taking it into account produces artifacts and considerably reduces the dynamic range. For a given field of view, or a given angular distance to the phase center, the importance of the W-term increases as the w-coordinate value, *i.e.* when the targeted field is at low elevation. It is therefore important to stress that wide fields of view or long baselines do not directly mean that the W-term will take an importance: whatever the baseline or field of view, a planar array that would observe at the zenith would always give a null w-coordinate.

Algorithmically, for A-Projection, the W-term is one of the main limiting factor. Using the W-Projection algorithm (Cornwell et al. 2008), assuming the W-term support is higher than the Prolate spheroidal support, the gridding time evolves as $t_{\text{grid},w} \propto w^2 \cdot D^4$, because the highest spatial frequency in the image plane has to be properly sampled. We found that on a typical LOFAR dataset this non-linear behavior generally makes the $\lesssim 5\%$ of the points with the highest w-coordinates to be responsible for $\gtrsim 70\%$ of the computing time (as in Fig. C.1). This little amount of data does not necessarily bring sensitivity or resolution. Setting a w_{max} value above which the visibilities are not gridded significantly increases the computational efficiency, without losing sensitivity or resolution.

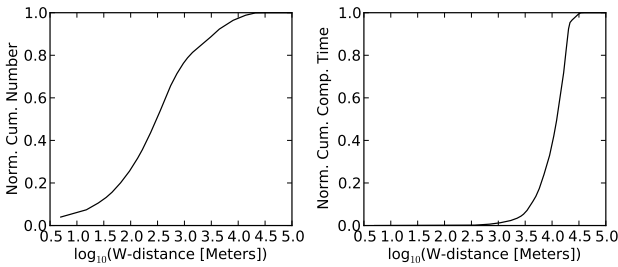


Fig. C.1. For a given field of view the W-term increases for lower elevation of the target field. Using the W-Projection algorithm, the computing time increases with w^2 . This figure shows the normalized cumulative distribution of the w-coordinate (left panel) for a typical LOFAR observation, and the corresponding normalized cumulative computing time (right panel). We can see that rejecting the $\sim 5\%$ of the points with $w > 10^4$ saves $\sim 70\%$ of the computing time.